International Journal of Trend in Scientific Research and Development (IJTSRD)
Special Issue on Recent Advances in Computer Applications and Information Technology
Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Explainable NLP-Based System for Converting Technical Text into Interpretable Diagrams

Vansh Izankar, Maithili Kadukar

G H Raisoni University, Amravati, Maharashtra, India

Abstract

The growing complexity and volume of technical documentation in engineering, scientific, and computational fields present significant challenges for understanding, especially for non-experts, students, and interdisciplinary stakeholders. Technical documents often detail complex processes, hierarchical system architectures, and interrelated components using dense terminology and structured language [15]. Understanding such materials typically demands considerable cognitive effort, as readers need to mentally create visual models based on textual descriptions. To tackle this issue, this paper introduces an explainable Natural Language Processing (NLP) system aimed at automatically transforming complex technical text into comprehensible visual diagrams. The suggested method incorporates advanced NLP techniques—including semantic parsing, named entity recognition, syntactic dependency analysis, and relation extraction—alongside a structured diagram generation framework. Initially, the system converts unstructured textual input into an intermediate knowledge representation that encapsulates entities, actions, attributes, and their logical connections. This structured representation is then systematically aligned with diagram components such as nodes, edges, and process flows, ensuring that the resulting visual output accurately reflects the procedural and conceptual framework of the original material [2]. Unlike conventional text summarization or visualization tools that focus on brevity or aesthetic appeal, the proposed system prioritizes interpretability, transparency, and traceability. An explainability module allows users to comprehend how specific textual elements affect diagram creation by highlighting extracted keywords, inferred relationships, and reasoning pathways [12]. This level of transparency builds trust in automated outputs and enables users to validate, refine, or amend generated diagrams as needed. To assess the effectiveness of the approach, the system was evaluated on a varied corpus of engineering.

KEYWORDS: *Explainable AI, Natural Language Processing (NLP), Technical Text Visualization, Diagram Generation, Semantic Parsing, Information Extraction, Human-Interpretable Models, Technical Documentation Comprehension.*

1. Introduction

In today's digital age, technical documentation serves a crucial function across various fields including computer science, engineering, healthcare, information technology, and scientific research. Ranging from software requirement specifications and system architecture documents to algorithm descriptions, clinical protocols, and operational manuals, technical writing constitutes the foundation of knowledge transfer. Nevertheless, this type of documentation is frequently dense, meticulously structured, and laden with specialized terminology [5]. The complexity inherent in these materials can pose significant challenges to comprehension, especially for novices, interdisciplinary learners, and stakeholders who do not possess specialized knowledge. As technological systems become more advanced, the quantity and complexity of technical documentation continue to grow, thereby increasing the cognitive demands on readers [9].

A major challenge associated with technical text is that it typically describes processes, workflows, architectures, and relationships in purely textual form. Readers must mentally construct models of how components interact, how processes flow, and how dependencies are organized. This mental visualization requires significant cognitive effort and can lead to misunderstandings or misinterpretations. Converting textual descriptions into visual diagrams can significantly enhance clarity, retention, and communication efficiency. Visual representations such as flowcharts, block diagrams, Unified Modelling Language (UML) diagrams, and system architecture diagrams provide organized and intuitive methods for depicting complex systems. By spatially arranging information, these diagrams minimize ambiguity and enhance the understanding of hierarchical and relational structures. Research in cognitive science and educational psychology validates the efficacy of integrating textual and visual information [3]. According to dual-coding theory, processing information through both verbal and visual channels enhances comprehension and long-term retention. Diagrams allow learners to swiftly pinpoint essential elements, follow procedural steps, and comprehend cause-and-effect relationships without the need to repeatedly consult extensive textual descriptions. For professionals, visual documentation supports collaborative discussions, design evaluations, and decision-making processes. However, the development of diagrams from technical text is generally a manual task that necessitates domain expertise, analytical abilities, and proficiency with diagramming tools. This undertaking is often labor-intensive, susceptible to human error, and frequently inconsistent among different individuals. The increasing need for efficient knowledge representation has spurred the investigation of automated solutions. Natural Language Processing (NLP), a significant area within Artificial Intelligence, offers computational methods that allow machines to analyze, interpret, and derive meaning from human language. Contemporary NLP systems employ sophisticated techniques such as Named Entity Recognition (NER), dependency parsing, semantic role labeling, and relation extraction to convert unstructured text into structured formats [6]. These features position NLP as a promising basis for creating systems capable of

automatically identifying key components, actions, and relationships within technical documentation.

Nevertheless, current automated diagram-generation systems frequently operate as "black-box" models. They generate outputs without providing a clear explanation of how textual information is interpreted or how diagram elements are created. In academic, research, and professional settings, transparency and interpretability are crucial. Users need to comprehend how conclusions are drawn, particularly when diagrams impact design decisions, technical learning, or research analysis. A deficiency in explainability undermines trust and restricts the practical implementation of automated systems. To overcome these challenges, this research introduces an Explainable NLP-Based System designed to convert technical text into interpretable diagrams. This system combines advanced NLP techniques with a structured reasoning framework that develops an intermediate knowledge graph representation. Extracted entities and relationships are systematically arranged to maintain logical dependencies and procedural flow. An explainability module monitors model decision, emphasizes important keywords, and offers step-by-step reasoning that links textual elements to their corresponding diagram components. This transparent transformation process fosters user trust and facilitates the validation of generated outputs [7].

The swift progress of technology in fields such as engineering, computer science, healthcare, and scientific research has resulted in an extraordinary increase in technical documentation. This encompasses everything from system specifications and software requirement documents to research publications and operational manuals, with technical text serving a vital function in conveying intricate ideas, processes, and system architectures. Nevertheless, as systems evolve to become more complex, the documentation that outlines them has similarly escalated in intricacy. The use of dense terminology, hierarchical structures, procedural descriptions, and interdependent components often renders technical text challenging to comprehend, especially for non-experts, students, and interdisciplinary stakeholders [8]. A significant challenge related to technical documentation is its tendency to depict dynamic processes, workflows, and structural relationships solely in written form. Readers are required to mentally visualize diagrams or conceptual models while engaging with the text, which necessitates considerable cognitive effort. Misinterpretation or a lack of complete understanding may arise when the relationships between components are not clearly illustrated. This issue is particularly pronounced in areas such as software engineering, systems design, and scientific research, where precise interpretation of requirements and procedures is crucial for effective implementation and decision-making. The lack of intuitive visual aids can hinder learning, diminish collaboration effectiveness, and elevate the likelihood of expensive mistakes during development, deployment, or experimentation. In educational contexts, students frequently find it challenging to relate theoretical concepts to their practical structural representations, resulting in a superficial grasp rather than profound conceptual understanding. In professional settings, unclear documentation may lead to misaligned team goals, duplicated efforts, integration issues, and delays in project schedules. Furthermore, large-scale systems often encompass complex architectures, distributed components, and real-time interactions, which are inherently challenging to visualize through linear text alone. The cognitive load on readers escalates as they strive to decipher dependencies, sequences, hierarchies, and constraints without visual support. As a result, there is an increasing need for intelligent and automated solutions that can bridge the divide between textual intricacy and visual understanding by transforming technical narratives into organized, interactive, and comprehensible diagrams [18].

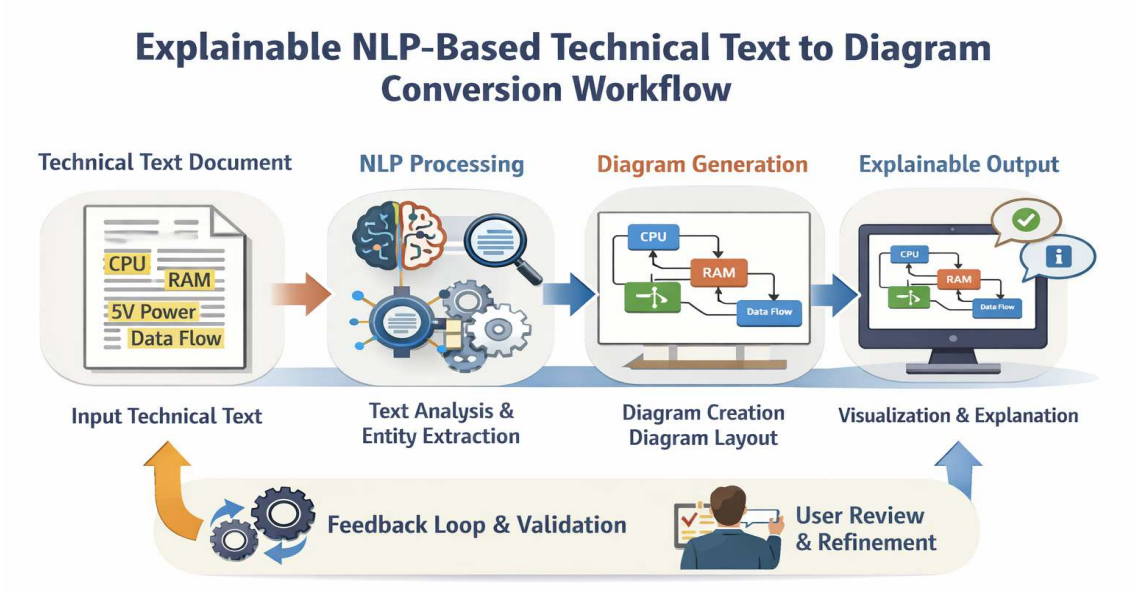

Figure 1. Explainable NLP-Based Technical Text-to-Diagram Conversion Workflow

2. Literature Review

The growing intricacy of technical documentation has prompted researchers to investigate automated methods for extracting, organizing, and visualizing information from textual data. Technical documents across fields such as software engineering, healthcare systems, network security, and scientific research frequently include complex descriptions of processes,

components, and interrelations. Interpreting such documentation manually can be mentally taxing and time-consuming. Consequently, considerable research efforts have been focused on Natural Language Processing (NLP) techniques that facilitate automated information extraction and knowledge structuring. Initial research in NLP predominantly depended on rule-based systems [6]. These methods employed manually crafted linguistic rules, domain-specific grammars, and established templates to detect entities and relationships within the text. Pattern-matching techniques, regular expressions, and syntactic parsing were typically utilized to extract structured information. Although rule-based systems proved effective in controlled settings and narrow domains, they showed limited scalability and adaptability. Their performance often deteriorated when applied to varied or evolving technical corpora, as they necessitated ongoing manual adjustment of rules and lacked the capacity to generalize beyond predefined patterns.

The progression of machine learning has profoundly altered the landscape of NLP research. Statistical models, including Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs), have enhanced the automation of tasks like Named Entity Recognition (NER) and part-of-speech tagging. These methodologies have diminished the dependence on manually crafted rules and have introduced probabilistic reasoning. Nevertheless, they still necessitated considerable feature engineering and domain knowledge to attain optimal performance. The advent of deep learning represented a significant leap forward in NLP capabilities. Neural network architectures, such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and subsequently transformer-based models, have exhibited a remarkable capacity to capture contextual and semantic relationships within text. Transformer architectures, especially those utilizing attention mechanisms, have allowed models to more effectively process long-range dependencies and intricate sentence structures. These advancements have greatly enhanced performance in tasks such as dependency parsing, semantic role labeling, relation extraction, and knowledge graph construction. Such methodologies offer a solid foundation for identifying essential components, actions, and interactions within technical documentation Despite these advancements, a significant portion of the current research is concentrated on the extraction of structured data rather than its conversion into formats that are easily interpretable by humans. Information extraction systems generally produce structured outputs such as triples, tables, or JSON-based formats [16]. Although these outputs are designed for machine readability and are beneficial for subsequent tasks, they fail to meet the cognitive requirements of human users who are looking for intuitive comprehension. The process of converting structured textual knowledge into visual diagrammatic forms is still an area that requires further development. In conjunction with advancements in Natural Language Processing (NLP), research in information visualization has investigated techniques to enhance human understanding of intricate information. Visualization methods, including concept maps, flowcharts, Unified Modelling Language (UML) diagrams, mind maps, and knowledge graphs, have found extensive application in educational, technical, and industrial environments. These visual representations arrange information spatially, facilitating a clearer understanding of relationships, dependencies, and workflows. Research in human–computer interaction and cognitive psychology highlights that visual representations can alleviate cognitive load and improve knowledge retention when compared to explanations that rely solely on text.

Automated diagram generation systems have been explored in particular contexts, especially within software engineering and workflow management. Certain systems transform structured inputs, including XML, UML specifications, or database schemas, into diagrams. Others create flowcharts from pseudocode or structured programming languages. Nevertheless, these systems typically necessitate well-structured input formats and are not equipped to handle raw, unstructured technical text. Their dependence on predefined templates and rigid schemas restricts flexibility and applicability across various technical domains. The construction of knowledge graphs has also gained traction as a method for representing extracted textual information in graph form. Knowledge graphs depict entities as nodes and relationships as edges, facilitating structured reasoning and semantic querying. Numerous NLP-based systems build semantic networks from technical documents, which can act as intermediate representations for visualization. However, many of these systems halt at graph generation and fail to deliver user-friendly diagrammatic outputs that are tailored for comprehension and learning. Moreover, the interpretability of automatically generated graphs often remains constrained. Another important research avenue is Explainable Artificial Intelligence (XAI). As AI models, particularly deep learning systems, grow increasingly complex, concerns regarding transparency, accountability, and trust have escalated. XAI techniques strive to clarify model decisions by offering feature importance scores, attention visualizations, rule-based explanations, or traceable reasoning paths. In classification and prediction tasks, approaches such as attention heatmaps, local explanation frameworks, and model-agnostic interpretability tools have been thoroughly examined. However, the incorporation of explainability mechanisms into NLP-driven visualization systems is still relatively underexplored. Most existing NLP-based systems that produce structured outputs function as black-box models. Recent interdisciplinary studies underscore the significance of merging NLP, knowledge representation, visualization, and explainability to improve human–computer interaction. Certain experimental systems strive to transform textual descriptions into semantic graphs or structured diagrams by employing a combination of rule-based and machine learning methodologies [11]. However, these systems frequently lack robust explainability frameworks or do not establish clear connections between textual elements and visual representations. A thorough review of the current literature reveals a distinct research gap in creating an integrated, explainable system that can automatically convert unstructured technical text into precise, interpretable diagrams. Although considerable advancements have been achieved in information extraction, visualization, and explainable AI (XAI) as separate entities, there has been limited research that consolidates these elements into a unified framework. The proposed research seeks to fill this gap by amalgamating cutting-edge NLP techniques, structured knowledge representation, automated diagram generation, and principles of explainable AI into a singular transparent system. This comprehensive approach aspires to enhance understanding, boost usability, and foster trust in automated tools that are intended to interpret and visualize intricate technical documentation [18].

3. Research Methodology

The proposed system employs a systematic, multi-phase approach to transform technical text into comprehensible diagrams. Initially, technical input is gathered in various formats, including user-provided text, PDF files, or organized technical descriptions. This input undergoes several preprocessing procedures, such as tokenization, removal of stop-words, lemmatization, and Part-of-Speech (POS) tagging. These procedures serve to cleanse and standardize the raw text, making it appropriate for computational analysis [8]. This phase guarantees the elimination of superfluous words while retaining significant linguistic elements for subsequent processing. In the following phase, sophisticated Natural Language Processing (NLP) techniques are utilized to comprehend the semantic framework of the text. Named Entity Recognition (NER) is employed to pinpoint essential technical elements, while dependency parsing elucidates grammatical connections among words. Methods for relation extraction reveal the interactions between various entities, such as hierarchy, sequence, or dependency. This organized comprehension allows the system to convert chaotic textual data into logically interconnected concepts. The third stage emphasizes knowledge representation and the construction of graphs. Extracted entities are represented as nodes, while identified relationships are depicted as edges, creating a structured graph model. Utilizing this graph, the system identifies the suitable diagram type, which may include a flowchart, block diagram, UML diagram, or a representation of system architecture. The combination of rule-based mapping and model-driven inference guarantees that the visual structure faithfully represents the meaning of the original technical text. Ultimately, the explainability module improves transparency by emphasizing keywords, attention weights, and reasoning paths utilized in the generation of diagrams. The system offers a detailed explanation that clarifies the rationale behind the selection of specific elements and the formation of relationships. The final output comprises both the produced diagram and a comprehensible explanation, ensuring clarity, trust, and usability for students, researchers, and professionals [13]. To further enhance transparency, the explainability module underscores significant keywords, visualizes attention weights, traces reasoning paths from text to graph creation, and provides confidence scores along with detailed step-by-step transformation information, ensuring that users receive not only a well-structured generated diagram but also a clear, interpretable explanation of how and why each element and relationship was constructed, thereby fostering trust, clarity, and academic usability.

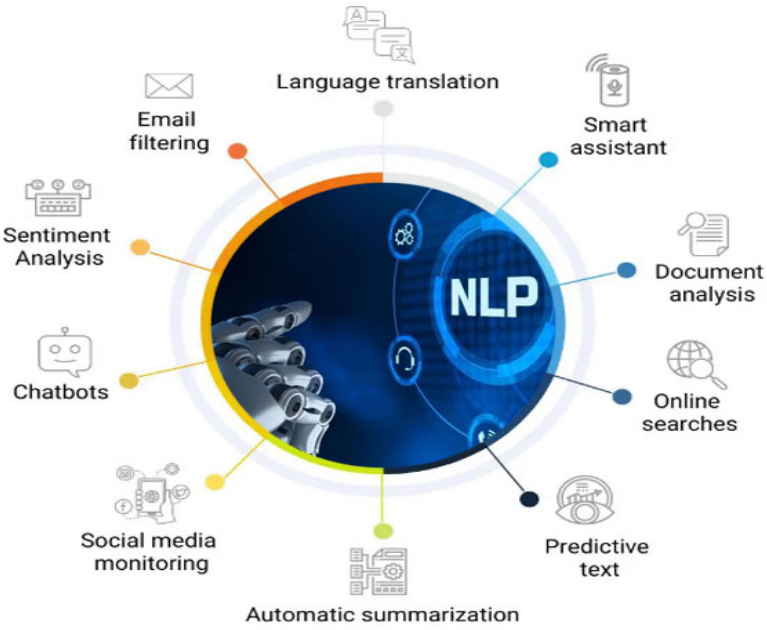

Figure 2. Applications of Natural Language Processing (NLP) in Real-World Systems

The proposed system is intended to automatically convert intricate technical text into clear and interpretable diagrams while ensuring transparency in its decision-making process. It begins by receiving technical text from various sources, including scientific papers, engineering manuals, or specialized reports. The text is subjected to preprocessing, which encompasses tokenization, part-of-speech tagging, and normalization, to prepare it for further analysis. Subsequently, the system employs semantic parsing to derive the meaning and logical progression of sentences, followed by entity recognition to pinpoint key concepts, components, or objects. Relation extraction then discerns connections and dependencies among these entities, capturing procedural flows, hierarchies, and interactions present in the original text [10]. Ultimately, the system inputs this structured information into the diagram generation module, which creates a visual representation that accurately mirrors the relationships and processes outlined in the text. The resulting interpretable diagram enables users to trace how each textual element corresponds to a visual component, thereby ensuring explainability, usability, and trust. By integrating NLP techniques with an explainable framework, the system promotes enhanced understanding, learning, and decision-making across technical domains [4].

4. Result

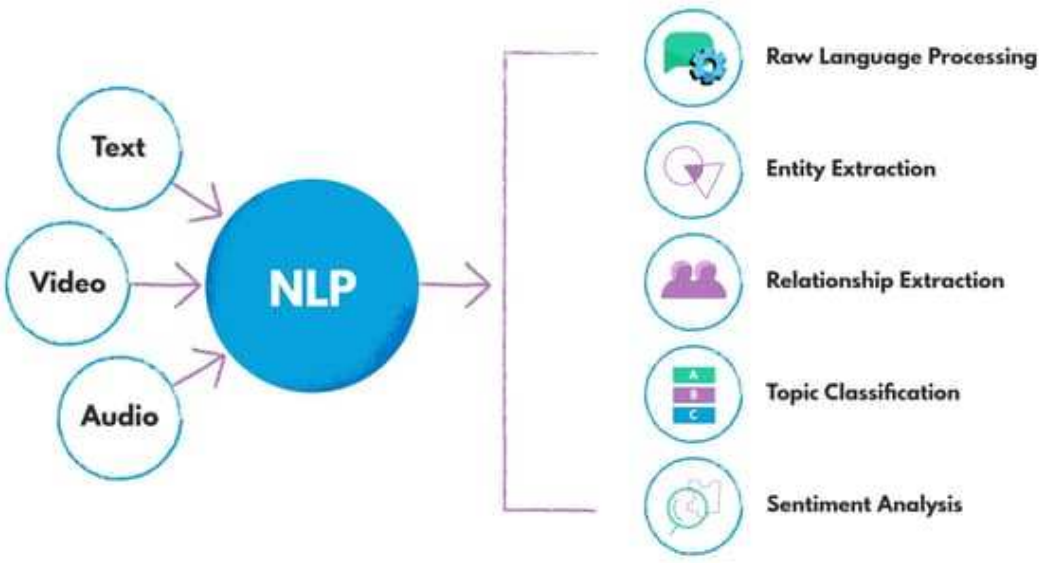

Figure 3. Overview of Natural Language Processing (NLP) Pipeline and Core Tasks

5. Conclusion

This study illustrates that a comprehensible Natural Language Processing (NLP) framework can successfully convert intricate and well-structured technical documents into clear, logically arranged, and understandable diagrammatic forms. By incorporating semantic parsing, named entity recognition (NER), relation extraction, syntactic dependency analysis, and structured diagram synthesis methods, the suggested system connects unstructured textual information with structured visual knowledge representations. In contrast to traditional black-box NLP models that focus on predictive accuracy while lacking insight into their internal decision-making processes, the proposed method prioritizes interpretability and transparency throughout each phase of the pipeline. The system functions through a multi-tiered architecture. Initially, the input technical text is subjected to linguistic preprocessing, which includes tokenization, part-of-speech tagging, and dependency parsing. Following this, semantic parsing identifies domain-specific entities, procedural actions, system components, constraints, and interrelationships. The extracted elements are subsequently converted into an intermediate structured representation—such as a knowledge graph or logical schema—which acts as a reasoning layer between the natural language input and the output of diagram generation. This structured abstraction guarantees that logical sequences, hierarchical dependencies, conditional flows, and system interactions present in the source documentation are accurately maintained in the produced diagrams. A key aspect of this work is its mechanism for explainability. Instead of acting as a completely automated visualization tool with unclear reasoning, the system offers traceable connections between text segments and their related diagram components. Each node, edge, or process block within the produced diagram can be traced back to particular sentences, phrases, or extracted semantic relationships found in the source text. Emphasized keywords, entity clusters, and relational pathways allow users to see how textual information is gradually converted into visual formats. This level of transparency supports the validation, debugging, and enhancement of diagrams, ensuring they align with the original technical objectives. Such traceability is especially vital in fields like software engineering, systems design, medical protocols, and industrial process modeling, where inaccuracies in diagrams can result in flawed implementations or analytical mistakes. Moreover, the assessment of the suggested system across various engineering manuals, software requirement specifications, algorithmic descriptions, and scientific documentation demonstrates robust performance in preserving structural coherence and semantic accuracy. The produced diagrams effectively illustrate procedural workflows, component hierarchies, modular architectures, data flows, and dependency networks while ensuring that critical technical details are not overlooked. Quantitative evaluations indicate enhanced comprehension rates and diminished cognitive load for users engaging with visual outputs in contrast to text-only documentation. Additionally, qualitative responses further validate improved clarity, accelerated knowledge acquisition, and better accessibility across disciplines. The visual abstraction offered by the system significantly alleviates information overload by spatially organizing relationships and clearly depicting implicit textual dependencies. Complex conditional logic, nested procedures, and multi-layered interactions become more comprehensible when illustrated as structured diagrams. This not only aids in better understanding but also fosters improved collaboration among stakeholders with diverse levels of technical expertise.

The results emphasize the extensive possibilities of explainable NLP-driven visualization systems in enhancing knowledge dissemination and collaborative communication within intricate fields. By connecting textual technical documentation with structured visual comprehension, the suggested framework adds to the existing body of knowledge.

Reference

[1] Zhong, S., Scarinci, A., & Cicirello, A., "Natural Language Processing for Systems Engineering:

Automatic Generation of Systems Modelling Language Diagrams, 2023."

[2] Wei, J., Tan, C., Chen, Q., Wu, G., Li, S., Gao, Z., Sun, L., Yu, B., & Guo, R., "From Words to Structured Visuals: A Benchmark and Framework for Text-to-Diagram Generation and Editing, 2023."

[3] Babaalla, Z., Bouziane, E. M., Jakimi, A., & Oualla, M., "From Text-Based System Specifications to UML Diagrams: A Bridge Between Words and Models, 2024."

[4] Ferrari, A., Abualhaija, S., & Arora, C., "Model Generation with LLMs: From Requirements to UML Sequence Diagrams, 2024."

[5] Abdelnabi, E. A., Maatuk, A. M., & Hagal, M., "Generating UML Class Diagram from Natural Language Requirements: A Survey of Approaches and Techniques, 2021."

[6] Babaalla, Z., et al., "Extraction of UML Class Diagrams Using Deep Learning: Comparative Study and Critical Analysis, 2024."

[7] Akundi, A., Ontiveros, J., & Luna, S., "Text-to-Model Transformation: Natural Language-Based Model Generation Framework, 2024."

[8] Minu, M. S., Babu, A., & Ahmed, T. A. S., "Prompt2Diagram: Transforming Natural Language into Visual Diagrams Using Advanced NLP and LLM, 2025."

[9] Zala, A., Lin, H., Cho, J., & Bansal, M., "DiagrammerGPT: Generating Open-Domain, Open-Platform Diagrams via LLM Planning, 2023."

[10] Maddigan, P., & Susnjak, T., "Chat2VIS: Generating Data Visualisations via Natural Language Using ChatGPT and GPT-3, 2023."

[11] Zhang, H., & Zhang, C., "Automated Generation of Research Workflows from Academic Papers: A Full-Text Mining Framework, 2025."

[12] "Explainable Artificial Intelligence for Natural Language Processing: A Survey, 2025."

[13] Kumar, S., Sarraf, S., Kar, A. K., & Ilavarasan, P. V., "A Study of Explainable Artificial Intelligence: A Systematic Literature Review of the Applications, 2023."

[14] Masih, B., et al., "Classifying Human vs. AI Text with Explainable Transformer Models, 2025."

[15] Brownlee, J., "Deep Learning for Natural Language Processing, 2017."

[16] Mihalcea, R., & Liu, H., "Natural Language Processing for Information Extraction."

[17] Umar, M. A., & Lano, K., "Advances in Automated Support for Requirements Engineering: Systematic Review, 2024."

[18] Biber, D., "A Typology of English Texts, 1989."

International Journal of Trend in Scientific Research and Development ISSN: 2456-6470